

Муниципальное автономное общеобразовательное учреждение  
Улу-Юльская средняя общеобразовательная школа Первомайского района

СОГЛАСОВАНА  
Педагогическим советом  
МАОУ Улу-Юльской СОШ  
(протокол от 31.08.2024 №1)

УТВЕРЖДЕНА  
Приказом директора  
МАОУ Улу-Юльской СОШ  
от 02.09.2024 № 111-О



ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ  
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА  
«Юный программист»

Возраст слушателей: 7-8 классы  
Срок реализации: 1 год

Автор-составитель:  
Маньлов В.А., учитель  
информатики

Улу-Юл 2024

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Данная рабочая программа внеурочной деятельности «Юный программист» предназначена для обучающихся 7-8 класса. Содержание обучения, представленное в программе курса «Юный программист», позволяет вести обучение школьников в режиме актуального познания. Практическая направленность курса на создание внешних образовательных продуктов блок-схем, алгоритмов, программ — способствует выявлению фактов, которые невозможно объяснить на основе имеющихся у школьников знаний.

Возникающие при этом познавательные переживания обуславливают сознательное отношение к изучению основных теоретических положений информатики.

В рамках предлагаемого курса изучение основ программирования на языке Python это не столько средство подготовки к будущей профессиональной деятельности, сколько формирование новых общеинтеллектуальных умений и навыков: разделение задачи на этапы решения, построение алгоритма и др.

### Цели:

- Понять значение алгоритмизации как метода познания окружающего мира, принципы структурной алгоритмизации;
- овладеть базовыми понятиями теории алгоритмов;
- сформировать навыки выполнения технологической цепочки разработки программ средствами языка Python;
- научиться разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python;
- научить применять навыки программирования на конкретной учебной ситуации;
- привить навыки проектной деятельности.

### Задачи:

#### *- обучающие*

- познакомить с понятиями алгоритма, вычислимой функции, языка программирования;
- научить составлять и читать блок-схемы
- сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;
- изучить основные конструкции языка программирования Python, позволяющие работать с простыми и составными типами данных (строками, списками, кортежами, словарями, множествами);
- научить применять функции при написании программ на языке программирования Python;
- научить отлаживать и тестировать программы, делать выводы о работе этих программ.

#### *- развивающие*

- способствовать расширению и развитию памяти, внимания, технического мышления, изобретательности;
- способствовать развитию алгоритмического мышления;
- способствовать формированию интереса к техническим знаниям;
- сформировать умение формулировать, аргументировать и отстаивать свое мнение;

- сформировать умение выступать публично с докладами, презентациями и т.п.

#### ***- воспитательные***

- воспитывать аккуратность и дисциплинированность при выполнении работы;
- способствовать формированию положительной мотивации к трудовой деятельности;
- способствовать формированию опыта совместного и индивидуального творчества при выполнении командных заданий;
- Формировать чувство коллективизма и взаимопомощи.

**Возраст слушателей:** обучающиеся 7 - 8 классов. Именно этому возрастному контингенту свойственна познавательная активность и любознательность.

**Местом проведения** занятий является МАОУ Улу-Юльская СОШ Первомайского района.

#### **Ожидаемые результаты освоения курса внеурочной деятельности**

##### *Личностные результаты*

К личностным результатам освоения информационных и коммуникационных технологий как инструмента в учёбе и повседневной жизни можно отнести:

- критическое отношение к информации и избирательность её восприятия;
- уважение к информации о частной жизни и информационным результатам других людей;
- осмысление мотивов своих действий при выполнении заданий с жизненными ситуациями;
- начало профессионального самоопределения, ознакомление с миром профессий, связанных с информационными и коммуникационными технологиями.

*Метапредметными результатами* изучения курса «Программирование в среде Scratch» являются формирование следующих универсальных учебных действий:

##### Регулятивные УУД:

- планирование последовательности шагов алгоритма для достижения цели;
- поиск ошибок в плане действий и внесение в него изменений.

##### Познавательные УУД:

- моделирование – преобразование объекта из чувственной формы в модель, где выделены существенные характеристики объекта (пространственно-графическая или знаково-символическая);
- анализ объектов с целью выделения признаков (существенных, несущественных);
- синтез – составление целого из частей, в том числе самостоятельное достраивание с восполнением недостающих компонентов;
- выбор оснований и критериев для сравнения, сериации, классификации объектов;
- подведение под понятие;
- установление причинно-следственных связей;
- построение логической цепи рассуждений.

##### Коммуникативные УУД:

- аргументирование своей точки зрения на выбор оснований и критериев при выделении признаков, сравнении и классификации объектов;
- выслушивание собеседника и ведение диалога;
- признание возможности существования различных точек зрения и права каждого иметь свою.

##### *Предметные результаты:*

- Узнать основные алгоритмические конструкции;
- Уметь составлять алгоритмы для решения прикладных задач;
- Уметь реализовывать алгоритмы на компьютере в виде программ

- Владеть основной терминологией в области алгоритмизации и программирования;

## СОДЕРЖАНИЕ КУРСА

### Тема 1. Знакомство с языком Python

Общие сведения о языке Python. Установка Python на компьютер. Режимы работы Python. Что такое программа. Первая программа. Структура программы на языке Python. Комментарии.

Практическая работа 1.1. Установка программы Python

Практическая работа 1.2. Режимы работы с Python.

Тест № 1. Знакомство с языком Python

### Тема 2. Переменные и выражения

Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова.

Выражения. Операции. Порядок выполнения операций.

Математические функции. Композиция.

Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран.

Пример скрипта, использующего ввод и вывод данных. Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами.

Практическая работа 2.1. Работа со справочной системой

Практическая работа 2.2. Переменные

Практическая работа 2.3. Выражения

Практическая работа 2.4. Задачи на элементарные действия с числами.

Тест № 2. Выражения и операции.

### Тема 3. Условные предложения

Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not). Условный оператор. Альтернативное выполнение. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.

Практическая работа 3. 1. Логические выражения

Практическая работа 3.2. "Условный оператор"

Практическая работа 3.3. Множественное ветвление

Практическая работа 3.4. " Условные операторы"

Самостоятельная работа № 1. Решение задач по теме " Условные операторы"  
Зачетная работа № 1. "Составление программ с ветвлением".

Тест № 3. "Условные операторы".

### Тема 4. Циклы

Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием. Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while. Обновление переменной. Краткая форма записи обновления. Примеры использования циклов.

Оператор цикла с параметром for. Операторы управления циклом. Пример задачи с использованием цикла for. Вложенные циклы. Циклы в циклах. Случайные числа.

Функция randrange. Функция random. Примеры решения задач с циклом.  
Практическая работа 4.1. "Числа Фибоначчи"

Практическая работа 4.2. Решение задачи с циклом for.

Практическая работа 4.3. Реализация циклических алгоритмов

Практическая работа 4.4. Случайные числа

Практическая работа 4.5. Решение задач с циклом.

Самостоятельная работа № 2 "Составление программ с циклом" Тест № 4. Циклы

Творческая работа № 1. "Циклы"

### **Тема 5. Функции**

Создание функций. Параметры и аргументы. Локальные и глобальные переменные. Поток выполнения. Функции, возвращающие результат. Анонимные функции, инструкция lambda. Примеры решения задач с использованием функций. Рекурсивные функции.

Вычисление факториала. Числа Фибоначчи.

Практическая работа 5.1. Создание функций

Практическая работа 5.2. Локальные переменные

Практическая работа 5.3. Решение задач с использованием функций

Практическая работа 5.4. Рекурсивные функции Самостоятельная работа № 3 по теме "Функции"

Тест № 5. Функции

### **Тема 6. Строки - последовательности символов**

Составной тип данных - строка. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки. Срезы строк.

Строки нельзя изменить. Сравнение строк. Оператор in. Модуль string. Операторы для всех типов последовательностей (строки, списки, кортежи). Примеры решения задач со строками.

Практическая работа 6.1. Строки

Практическая работа 6.2. Решение задач со строками.

### **Тема 7. Сложные типы данных**

Списки. Тип список (list). Индексы. Обход списка. Проверка вхождения в список.

Добавление в список. Суммирование или изменение списка.

Операторы для списков. Срезы списков. Удаление списка. Клонирование списков.

Списочные параметры. Функция range. Списки: примеры решения задач.

Матрицы. Вложенные списки. Матрицы. Строки и списки. Генераторы списков в Python.

Кортежи. Присваивание кортежей. Кортежи как возвращаемые значения

Введение в словари. Тип словарь (dict). Словарные операции. Словарные методы.

Множества в языке Python. Множества. Множественный тип данных. Описание множеств. Операции, допустимые над множествами:

объединение, пересечение, разность, включение. Оператор определения принадлежности элемента множеству.

Практическая работа 7.1. Списки.

Практическая работа 7.2. Решение задач со списками. Тест № 7. Списки

### **Тема 8. Стиль программирования и отладка программ.**

Стиль программирования. Отладка программ. Зачет по курсу «Юный программист»

## **УЧЕБНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ**

Наименование тем	Кол-во часов, всего
<b>Тема 1. Знакомство с языком Python</b>	<b>1,5</b>
Урок 1. Общие сведения о языке	0,5
Практическая работа 1.1. Установка программы Python	
Урок 2. Режимы работы	1
Практическая работа 1.2. Режимы работы с Python	
Тест № 1. Знакомство с языком Python	
<b>Тема 2. Переменные и выражения</b>	<b>4,5</b>
Урок 3. Переменные	1
Практическая работа 2.1. Работа со справочной системой	
Практическая работа 2.2. Переменные	
Урок 4. Выражения	1
Практическая работа 2.3. Выражения	
Практическая работа 3.2. "Условный оператор"	
Урок 5. Ввод и вывод	1
Урок 6. Задачи на элементарные действия с числами	1,5
Практическая работа 2.5. Задачи на элементарные действия с числами	
Тест № 2. Выражения и операции	
<b>Тема 3. Условные предложения</b>	<b>7,5</b>
Урок 7. Логические выражения и операторы	1,5
Практическая работа 3. 1. Логические выражения	
Урок 8. Условный оператор.	1,5
Практическая работа « 3.2. «Условный оператор»	
Урок 9. Множественное ветвление.	1,5
Практическая работа №3.3. «Множественное ветвление»	
Урок 10. Реализация ветвления в языке Python	1,5
Практическая работа 3.4. Условные операторы"	
Самостоятельная работа № 1 по теме "Условные операторы"	
Урок 11. Зачетная работа № 1. "Составление программ с ветвлением" .	1,5
Тест № 3 «Условные операторы».	
<b>Тема 4. Циклы</b>	<b>10,5</b>
Урок 12. Оператор цикла с условием.	1,5
Практическая работа 4.1. "Числа Фибоначчи"	
Урок 13. Оператор цикла for	1,5
Практическая работа 4.2. Решение задачи с циклом for.	
Урок 14. Вложенные циклы	1,5
Практическая работа 4.3. Реализация циклических алгоритмов	
Урок 15. Случайные числа. Практическая работа № 4.4. «Случайные числа»	1,5
Урок 16. Пример решения задач с циклом	1,5

Практическая работа 4.5. Решение задач с циклом.	
Самостоятельная работа № 2 «Составление программ с циклом»	
Тест № 4. Циклы	1,5
Урок 17. Творческая работа № 1. "Циклы"	1,5
<b>Тема 5. Функции</b>	<b>9</b>
Урок 18. Создание функций	1,5
Практическая работа 5.1. Создание функций	
Урок 19. Локальные переменные	1,5
Практическая работа 5.2. Локальные переменные	
Урок 20. Примеры решения задач с использованием функций	3
Практическая работа 5.3. Решение задач с использованием функций	
Самостоятельная работа № 3 по теме "Функции"	
Урок 21. Рекурсивные функции	3
Практическая работа 5.4. Рекурсивные функции	
Тест № 5. Функции	
<b>Тема 6. Строки- последовательность символов</b>	<b>4,5</b>
Урок 22. Строки	1,5
Урок 23. Срезы строк	1,5
Практическая работа 6.1. Строки	
Урок 24. Примеры решения задач со строками	1,5
Практическая работа 6.2. Решение задач со строками.	
<b>Тема 7. Сложные типы данных</b>	<b>10,5</b>
Урок 25. Списки	1,5
Урок 26. Срезы списков	1,5
Практическая работа 7.1. Списки	
Урок 27. Списки: примеры решения задач	1,5
Практическая работа 7.2. Решение задач со списками	
Урок 28. Матрицы	1,5
Тест № 7. Списки	
Урок 29. Кортежи	1,5
Урок 30. Введение в словари	1,5
Урок 31. Множества в языке Python	1,5
<b>Тема 8. Стиль программирования и отладка программ</b>	<b>3</b>
Урок 32. Стиль программирования	0,5
Урок 33. Отладка программ	0,5
Урок 34. Зачет по курсу «Программирование на языке Python»	2
<b>ВСЕГО</b>	<b>51</b>

## ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

1. Домашняя страница Python [www.python.org](http://www.python.org). Справочные материалы, официальная документация.
2. Сайт проекта Институт: Национальный открытый университет, курс «Введение в программирование на Python», <http://www.intuit.ru/studies/courses/12179/1172/info>.
3. Сайт проекта Институт: Национальный открытый университет. Курс «Язык программирования Python» <http://www.intuit.ru/studies/courses/49/49/info>.
4. Сайт проекта Open Book Project [openbookproject.net](http://openbookproject.net) содержит серию практических примеров на Python Криса Мейерса.
5. Python. Подробный справочник Дэвида М. Бизли — книга со справочной информацией о языке Python и модулях стандартной библиотеки.
6. Python. Справочник Марка Лутца. Справочник по наиболее часто используемым функциям и модулям.